DOCUMENT RESUME

ED 046 067 EA 003 186

AUTHOR Rumble, Richard T.
TITLE A Computer Solution of the Parking Lot Problem.
PUB DATE 28 Oct 70
NOTE 19p.; Paper presented at Operations Research Society of America National Meeting (38th, Detroit, Michigan, October 28-30, 1970)

EDRS PRICE EDRS Price MF-$0.65 HC-$3.29
DESCRIPTORS Computer Graphics, *Computer Programs, *Design, *Design Needs, Geometric Concepts, *Parking Areas, Parking Controls, Traffic Circulation

ABSTRACT

A computer program has been developed that will accept as inputs the physical description of a portion of land, and the parking design standards to be followed. The program will then give as outputs the numerical and graphical descriptions of the maximum-density parking lot for that portion of land. The problem has been treated as a standard optimization one in which the number of parking stalls are to be maximized subject to the constraint that a statistically representative vehicle could easily maneuver in the lot. Dynamic programing was used to obtain the optimum solution. (Author)

EDO 46067

A Computer Solution of the Parking Lot Problem*

by

Richard T. Rumble**

U.S. DEPARTMENT OF HEALTH, EDUCATION & WELFARE
OFFICE OF EDUCATION

THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL OFFICE OF EDUCATION POSITION OR POLICY.

Paper presented for 38th National ORSA Meeting, on 28 October 1970.

EA 003 196

* This work was supported in part by a United States Air Force Office of Scientific Research Grant AFOSR Grant 699-67.

** Department of Electrical Engineering
University of California, Santa Barbara
Santa Barbara, California 93106

TABLE OF CONTENTS

ABSTRACT	*ii*
LIST OF ILLUSTRATIONS	*iv*
INTRODUCTION	1.
PROBLEM STATEMENT	2.
BACKGROUND	2.
DEVELOPMENT	6.
FUTURE WORK	15.
CONCLUSIONS	15.
BIBLIOGRAPHY	16.

ILLUSTRATIONS

FIGURE NO.		Page
1.	Required Width versus Parking Angle	4.
2.	Corrected Width versus Angle Curves	5.
3.	Flowchart of Dynamic Programming Algorithm	8.
4.	Flowchart of Integer Programming Algorithm	10.
5.	Aisle Configuration for Plan Three	12.
6.	Aisle Configuration for Plan Four	12.
7.	Aisle Configuration for Plan Five	12.
8.	Comparison of Designs	14.

INTRODUCTION

In the twenty-five years following the end of World War II, the automobile has risen in importance in the lives of people throughout the world and especially in the United States. The automobile has a prominent position in the western culture, but this prominence has led to problems in dealing with the volume of automobiles in daily use. The automobile is no longer used primarily for pleasure; instead it is used for commuting and shopping. Since each automobile seldom carries more than one or two passengers, a considerable portion of land is required for parking at every traffic generator. The convenience of the automobile is reduced if much time and effort must be devoted to finding a parking space, or if the available parking is not near to the desired destination.

As the number of registered vehicles and licensed drivers continues to grow, the design of efficient parking facilities becomes of considerable importance. Parking lot design in the past has been in the domain of architects and traffic engineers who, for the most part, have done a creditable job. However, their methods of parking-lot design depends upon the use of empirical data and rules of thumb obtained from years of unsystematic trial and error graphical design, an adequate method for small lots but inadequate when applied to large lots. The purpose of this paper is to discuss a method which mechanizes the design process and at the same time applies a systems-theoretic approach to the problem of parking-lot design.

PROBLEM STATEMENT

The problem of parking-lot design can be formulated as a classic optimization problem and in fact it belongs to the class known as knapsack problems. Given a limited resource, namely a portion of land of a certain size and shape, allocate that resource in an optimum manner by maximizing the number of parking stalls within the given portion of land subject to the constraint that the placement of the stalls conforms to a predefined set of design standards. These standards determine the ease with which a typical vehicle can maneuver through the lot when guided by a typical driver. Each stall is assumed to be randomly accessible, as in a self-service public lot. An additional constraint is implicit in this problem and that is the number of stalls must always be an integer.

A practical solution to this problem consists of a computer program that accepts as input the dimensions of the lot and the necessary information regarding the design standards to be used, determines the optimum design, and presents the results either in a tabular or visual format, or both.

BACKGROUND

Very little has been written on the systematic design of parking lots. The first publications appeared around 1948 and these have been followed by occasional publications. In 1956, the Bureau of Public Roads published a "Parking Guide for Cities"[1] which dealt with many aspects of parking, but gave only a cursory treatment to parking lot design. The following year, however, saw the publication

of "Traffic Design of Parking Garages", by E. R. Ricker[2] which presented formulas relating to the maneuverability of a vehicle as a function of certain vehicle dimensions. In 1966, the Department of Housing and Urban Development sponsored a large research program entitled, "New Systems Study in Urban Transportation". None of the studies in that program dealt with parking-lot design.

The publication that provides the best basis of information relating to lot design is "A Technical Report on Parking Design Requirements for Off-street Parking," by C. L. Lefler.[3] In this report, Lefler describes how the dimensions of a design vehicle were determined from the cumulative distributions of automobiles manufactured in the U.S. from 1962 to 1966. By applying these dimensions of a composite design vehicle to the formulas given by Ricker, he generated curves that show the width required for angled parking for angles of 30° to 90°.

Figure 1 shows these curves and the four types of aisles used. The W_1 aisle is a single-loaded aisle since there are stalls on only one side of the travel lane. The other three aisles, W_2, W_3, and W_4, are double-loaded aisles, with stalls on both sides of the travel lane. They differ only in the way of measuring the width. A few errors and inconsistencies were found in these curves, so one curve that appeared correct, the W_2 curve, was picked as a reference from which the other three were determined. These corrected curves appear in Fig. 2.

These curves then become the basis for the actual design of the parking lots. They contain implicitly all requirements of vehicle maneuverability and level of service. Changes in the design vehicle or desired level of service, should they occur, would be

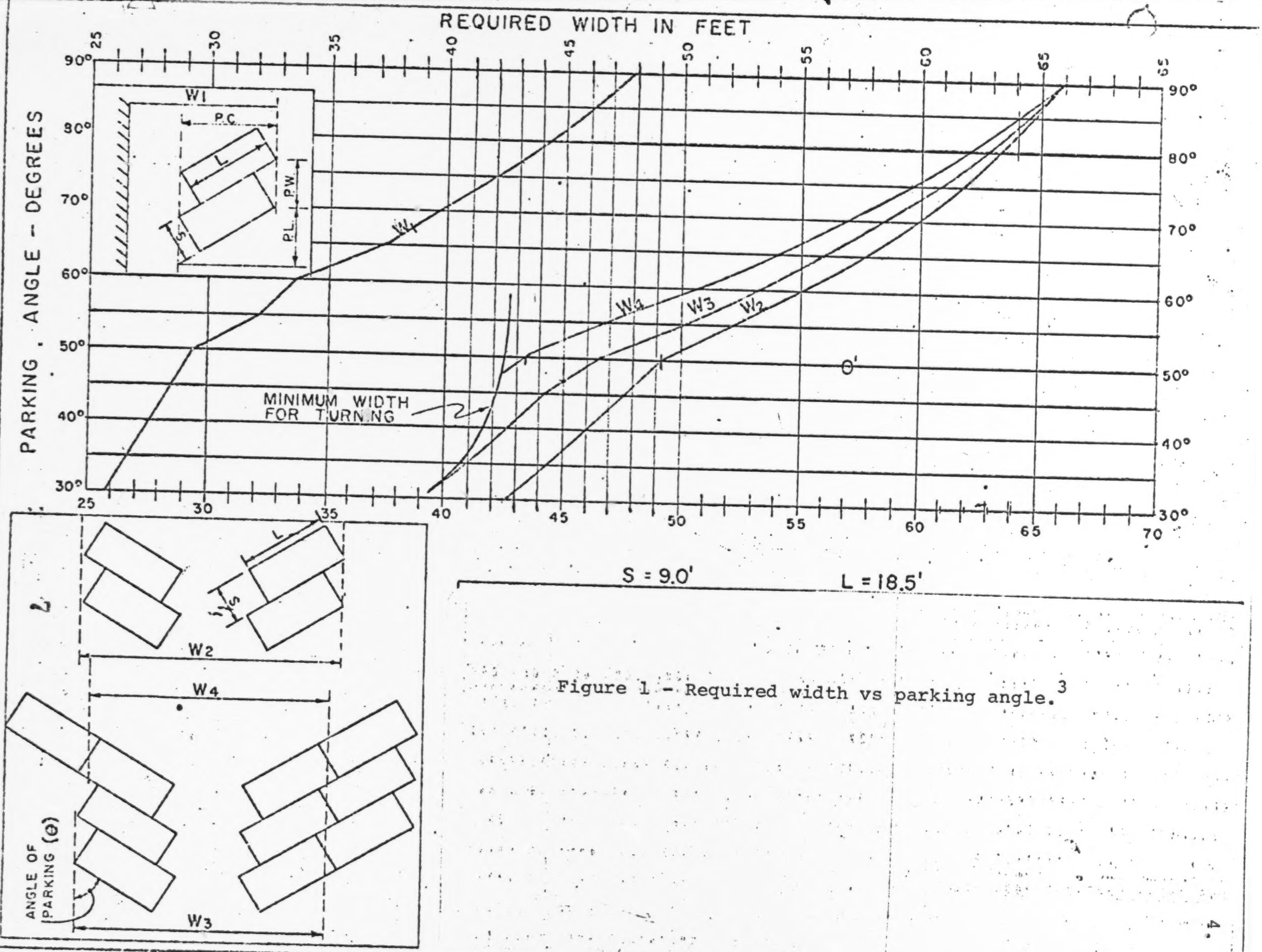


Figure 1 - Required width vs parking angle.[3]

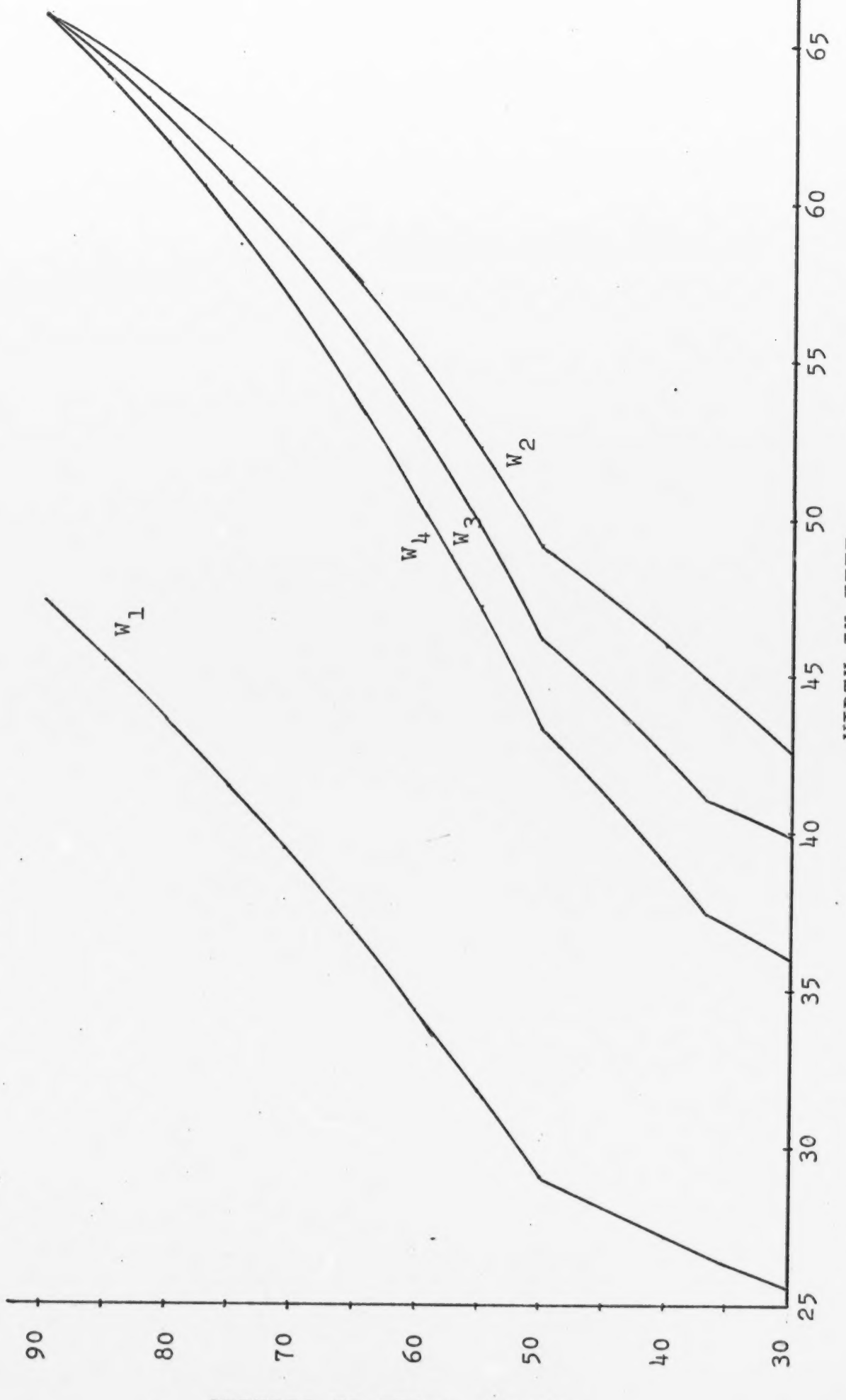


Figure 2 - Corrected width vs angle.

reflected in changes in the width vs angle curves given above, but would not affect the form of the design procedure.

DEVELOPMENT

Parking lots vary in size from a few stalls at the local drive-in restaurant to the open fields around the Ontario Motor Speedway. In urban areas there are predominantly two types of lots: small municipal or private lots with dimensions on the order of 100 feet or so, and those lots associated with shopping centers and stadia that cover an area equal to one or more city blocks. Both types utilize angle parking, with angles usually between 40° and 90°. Once certain decisions regarding the level of service desired for a particular lot have been made, the only variables remaining are the angle or angles to be used, and the orientation of the aisles.

Since most parking lots are either rectangles or combinations of rectangles, a natural first step toward the solution of the general lot design problem is to have a program which designs rectangular lots only. As a means to this end, a subordinate problem, worthy of consideration in its own right, was investigated, that of an infinite strip, i.e. a lot having a finite width and infinite length. In this case, the optimum design is the aisle assignment that maximizes the number of stalls per unit length of the strip. Clearly the optimum design of a rectangular lot whose length is much greater than its width approaches the infinite strip solution of the same width. Moreover, since this solution optimizes the stalls per running foot of the strip then it also

is optimum for the central portion of a rectangular lot.

The density of stalls per unit length is obtained from the relation $d(\theta_i) = PW^{-1}(\theta_i)$ for single loaded aisles. The density for double loaded aisles is $d(\theta_i) = 2 \cdot PW^{-1}(\theta_i)$. So for every angle on the menu, there is a corresponding width and density obtained from the above relations and the curves of Fig. 2. Two programs were written that solved the infinite strip problem for widths of 30 feet to 300 feet in increments of 1 foot. One program used a dynamic programming algorithm, the other an integer programming algorithm.

The dynamic program determined optimum solutions with a menu of five angles. Mathematically, the problem is to find

$$f_i(z) = \max_{x_i} \{d_i x_i + f_{i-1}(W - w_i x_i)\}$$

$$\text{for} \qquad x_i \leq [\frac{W}{w_i}]$$

where

x_i is the number of aisles of the i^{th} type.

d_i is the density of the i^{th} aisle.

w_i is the width of the i^{th} aisle.

W is the width of the infinite strip.

f_{i-1} is the return from the $i-1^{th}$ stage.

The menu used only double loaded aisles. Another set of solutions was obtained with a menu of 10 selections: the five used previously and five more that were single loaded aisles with the same angles. Figure 3 shows the flow chart for the dynamic programming algorithm used.

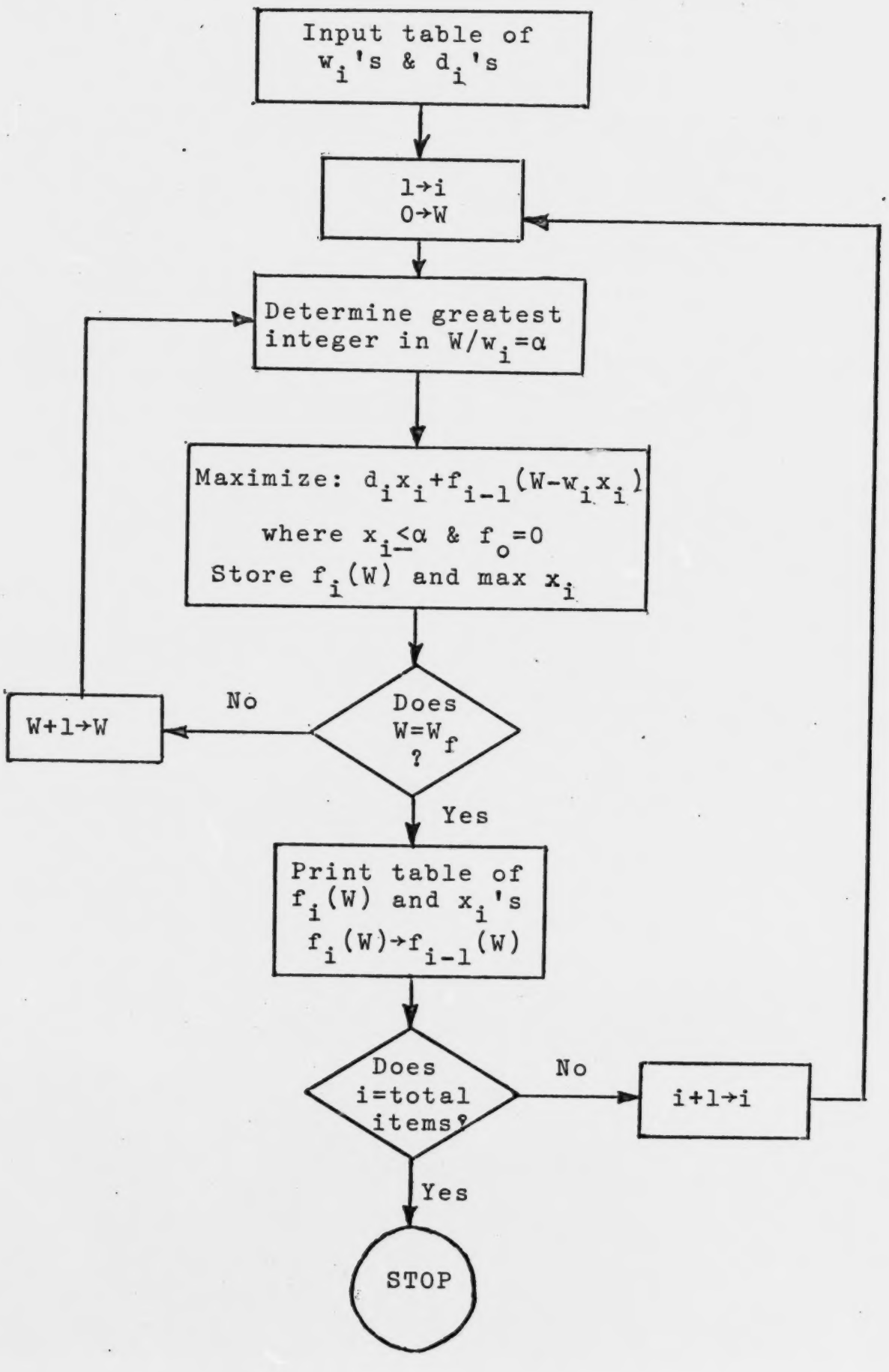


Figure 3 - Flow chart of dynamic programming algorithm.

The integer programming algorithm used in the other program is due to Gomory (4). The problem is stated mathematically as find z such that

$$z = \underset{\substack{x_i \\ i=1,2,\ldots,N}}{\text{maximum}} \sum_{i=1}^{N} d_i x_i$$

subject to the constraints that

$$\sum_{i=1}^{N} w_i x_i \leq W$$

and the x_i's are non-negative integers.

where

z is the optimum feasible solution given W.

W is the width of the infinite strip.

x_i is the number of aisles of the i^{th} type.

d_i is the density of the i^{th} aisle.

w_i is the width of the i^{th} aisle.

A flow chart of this integer programming algorithm is given in Fig. 4. Five solutions were obtained from this program for the widths of 50 feet to 250 feet. The menu was the same as used for the first set of solutions by dynamic programming. Only five cases were considered since the integer program was about 25 times slower than the dynamic program. Apparently this is not an uncommon problem with integer algorithms (4). Also the integer program failed to converge in one case. For these reasons, integer programming was dropped in favor of dynamic programming.

A second dynamic program was written which utilized a different menu and which was staged differently from the previous

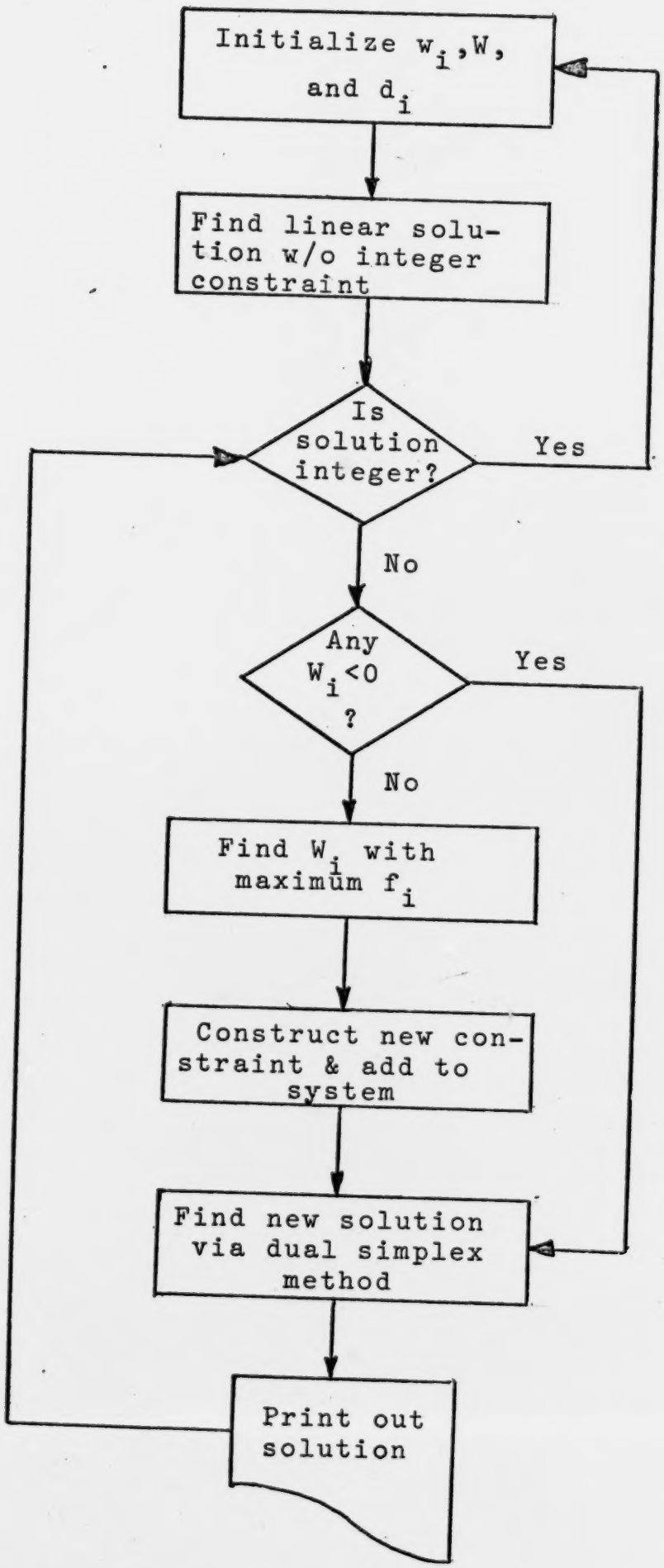


Figure 4 - Flow chart of integer programming algorithm.

program. The new menu contained 41 selections for widths between 26 feet and 66 feet, thus including both single and double loaded aisles. The first stage of the solution allowed selection of any one angle from the menu at each width. The second stage allowed the selection of two different angles, and so on through five stages. As before the width increment was one foot over the range of widths from 30 feet to 300 feet.

The policy function generated by this program produced an interesting result. No more than three different angles are used at any particular width and usually only one or two angles are picked. Thus anytime the staging of a dynamic program is by the number of different angles allowed, one need not provide for a choice of four or more different angles.

The next step was to try a new formulation of the infinite strip problem. A new dynamic program was written to solve for the optimum angle assignments. In this new approach, the angles on the menu will be those from 30° to 90° in 5° increments for a total of 13. Now, the strip widths from 26 feet to 39 feet, the optimum solution is taken from the W_1 curve, and for widths from 39 to 65 feet, from the W_2 curve. The next stage is to combine a W_1 aisle and a W_4 aisle to obtain a strip with three rows of stalls as shown in Fig. 5. Notice that θw_1 must be equal to or greater than θw_4 for a vehicle to be able to enter or leave a center stall in one motion. Since there are three rows of stalls in this configuration, it is called plan three. The optimum density is then obtained for each increment of width.

The next stage, plan four, is obtained by adding a row of

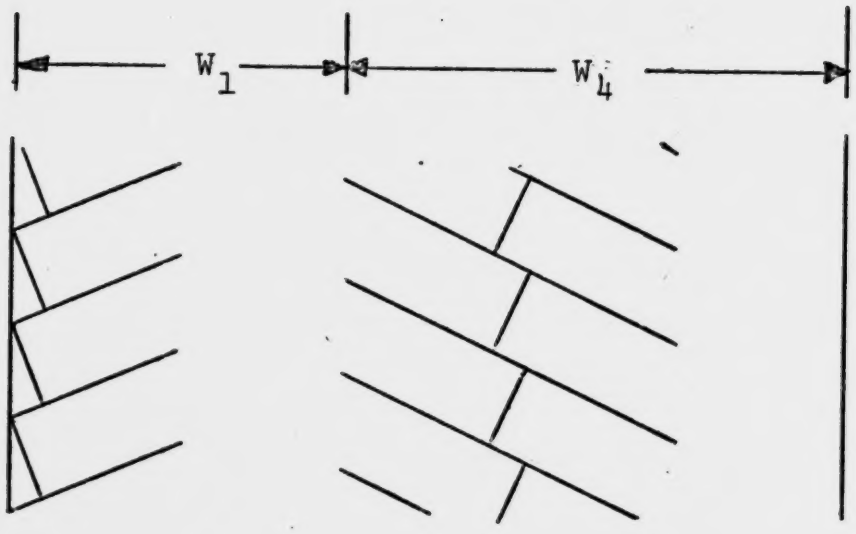


Figure 5 - Aisle configuration for plan three.

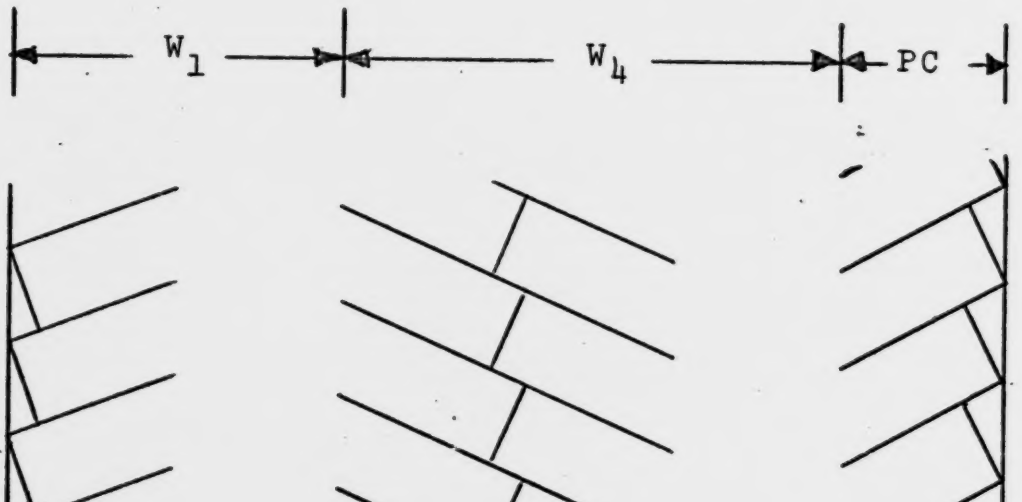


Figure 6 - Aisle configuration for plan four.

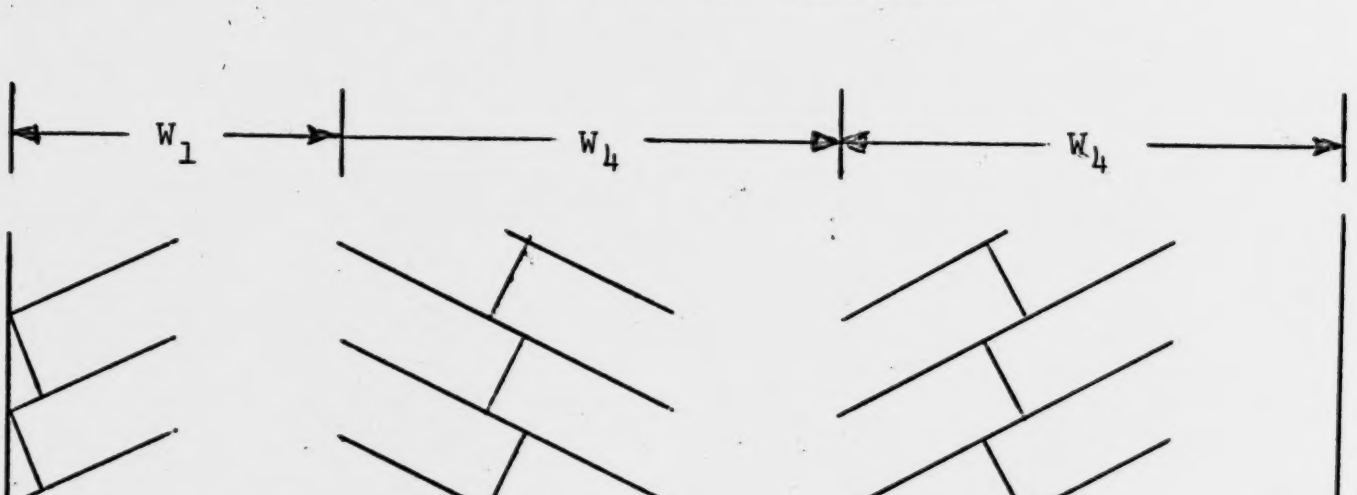


Figure 7 - Aisle configuration for plan five.

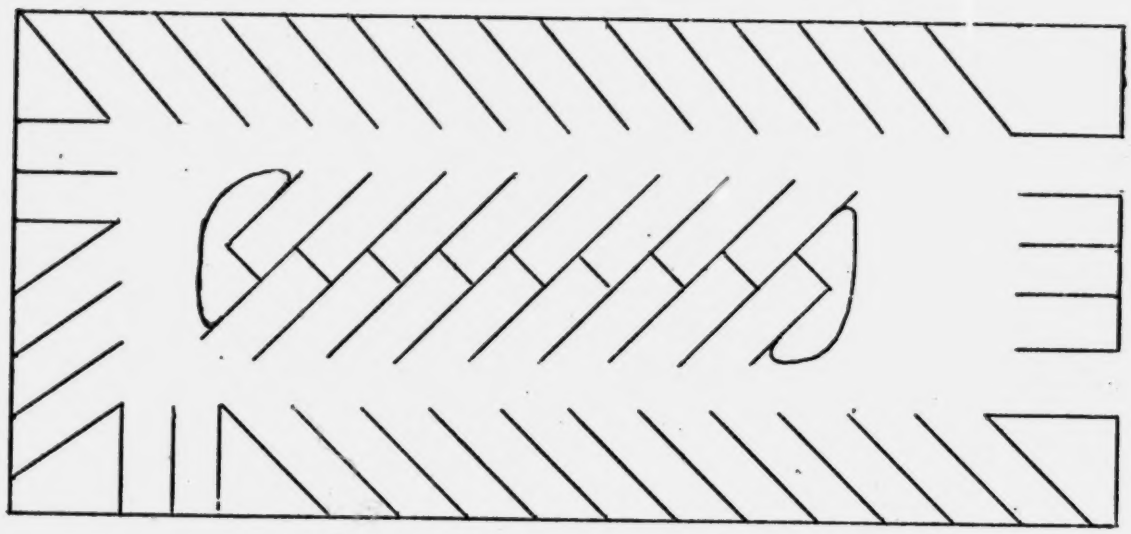
stalls to plan three as in Fig. 6. In this case, θ_{w4} must be equal to or greater than θ_{pc} for the same reason as the previous angle constraint. Again the optimum return is found at each increment of width. Plan five is a combination of plan three and an additional W_4 aisle. Plan six is obtained from plan five in the same manner as plan four was obtained from plan three. This process was continued until the optimum solutions were obtained for widths from 65 feet to 300 feet.

A new program is now under development which will use the policy function from the "plan" program to design a rectangular lot. This program will be a mechanization of the methods outlined in Lefler's report, but instead of using his criteria for choosing the appropriate lot plan, it will use the policy function described above.

Two examples are present here to show the expected performance of this program. Considering first a lot that is 100 feet by 200 feet, both Lefler's criteria and the plan policy function indicate four rows of stalls each using an angle of 55°, hence the program would give the same result as Lefler's report, or 56 stalls.

If the lot is 90 feet by 200 feet, Lefler's criteria choose equal angles of 45°. The plan policy function indicates three rows of 45° stalls and one of 50° stalls. The first case results in a total of 49 stalls, the latter in a total of 51 stalls. Figure 8 shows the two designs for comparison.

Dynamic Programming - 51 Stalls



Rule of Thumb - 49 Stalls

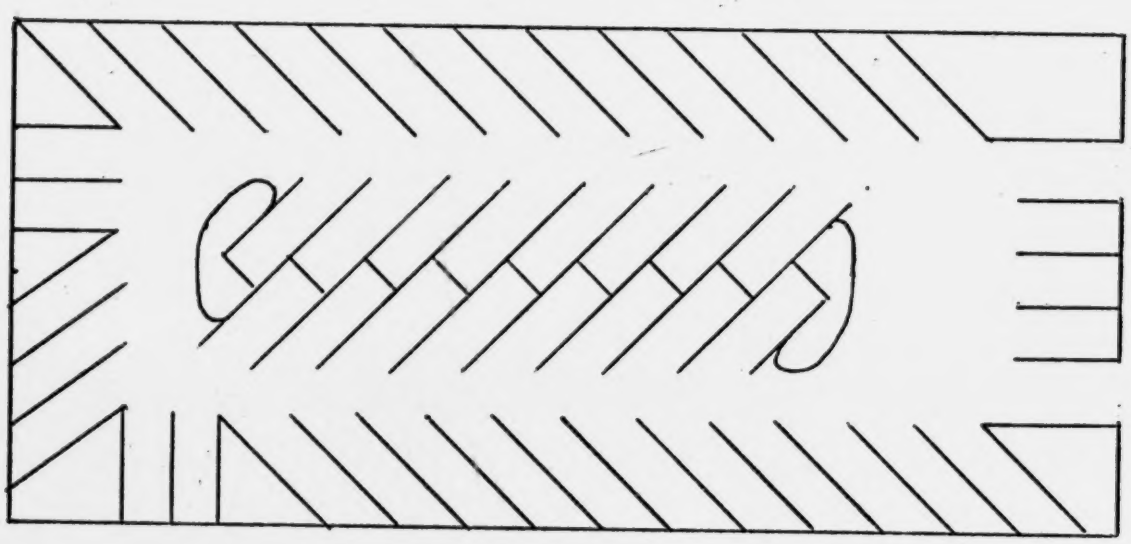


Figure 8 - Comparison of designs.

FUTURE WORK

In the very near future the development work will be completed on the rectangular lot design program. The major obstacle for the development now is choosing the proper criteria for treating the ends of the lots. When the program is working it will be validated by comparing designs with those that would be made via Lefler's report and with existing lots in the Santa Barbara area.

An additional program will be developed that will design a large parking lot as for a shopping center or a stadium. This program will determine the optimum design for rectangular lots as well as lots made up of a combination of rectangles. The latter will probably best be treated by designing for the individual rectangular shapes and then interfacing them through some suitable criteria. Since no published work treats this large scale design problem, the program designs would be validated by comparing them with existing lot designs.

These programs will include subroutines for graphically displaying the optimum parking lot design.

CONCLUSIONS

It has been shown that some improvement in parking lot design is possible through the use of a dynamic programming algorithm in a computer program. Equivalent or greater improvements are expected in dealing with the design of large parking lots.

BIBLIOGRAPHY

1. Bureau of Public Roads, Parking Guide for Cities, U.S. Government Printing Office, Washington, 1956.

2. Edmund R. Ricker, The Traffic Design of Parking Garages, Eno Foundation, Saugatuck, 1957.

3. Clinton L. Lefler, "A Technical Report on Parking Design Requirements for Off-street Parking." Santa Barbara, 1967.

4. Hillier, Frederick S. and Lieberman, Gerald J., Introduction to Operations Research, Holden-Day, San Francisco, 1967.